(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 240 667 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **87101841.2**

(51) Int. Cl.4: **G06F 15/16**

(22) Date of filing: **10.02.87**

(30) Priority: **12.03.86 JP 52448/86**
20.10.86 JP 247436/86

(43) Date of publication of application:
14.10.87 Bulletin 87/42

(84) Designated Contracting States:
DE FR GB

(71) Applicant: **HITACHI, LTD.**
6, Kanda Surugadai 4-chome
Chiyoda-ku Tokyo 101(JP)

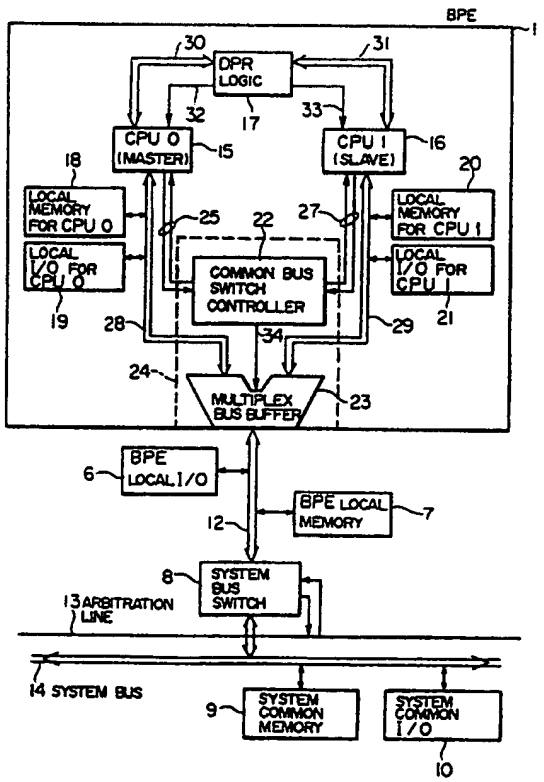
(72) Inventor: **Kametani, Masatsugu**
Nihariryo 3602, Shimoinayoshi Chiyodamura
Nihari-gun Ibarai-ken(JP)

(74) Representative: **Patentanwälte Beetz sen. - Beetz jun. Timpe - Siegfried - Schmitt-Fumian**
Steinsdorfstrasse 10
D-8000 München 22(DE)

(54) Processor.

(57) A processor for constructing a single processor system or multiprocessor system comprises, within a base processor element (1) constituting the processor, two CPUs (15, 16) with associated local memories (18, 20), a dual-port RAM (17) accessible from said CPUs, and a common bus switch circuit (22) for connecting any one of said CPUs to a common bus (12) shared by said CPUs.

FIG. 1



EP 0 240 667 A2

PROCESSOR

BACKGROUND OF THE INVENTION

This invention relates to a processor suitable for constructing a single processor system or multiprocessor system.

A conventional multiprocessor system, as shown for example in U.S. Patent No. 4,494,188, consists primarily of a CPU, a memory, and a common bus switch which operates on a master-slave basis with other processor elements. Such a multiprocessor system configured by single-CPU processor elements operates favorably so far as a customized task process under less disturbances is concerned. However, as the process charged on the system becomes more sophisticated, such as the case of the intellectualized control process, overhead system supports including the management of the data base and system status, the organization of the intelligent processing system based on the data base or sensor information, multi-interrupt processing, and multijob functions are rendered indispensable, and in general these processes are programmed by a high grade language and executed under a high grade operating system with the realtime multitasking and multijob supporting abilities.

In the above-mentioned conventional multiprocessor system, the realtime control process which is the main object of speedup is treated as one of tasks carried out by multitasking, and therefore at present the system cannot implement a tight-linked parallel processing due to the task switching overhead and disordered parallel processing schedule.

On this accout, the intelligent processing system is often separated from the parallel-processed control processing system by employment of a super minicomputer as a supervisory system, but in this case, there are several problems as follows;

(I) the communication between the control processing system and intelligent processing system is apt to become less frequent, therefore, the advantages of distributed intelligent processing and system management which generally necessitate operating system overhead for controlling the internal status of each own processor are not effectively exerted, resulting in a degraded cost-effectiveness evaluation, and

(2) it is difficult to expand the processing ability of the intelligent processing system to match the expansion of the processing ability of the control processing system and also difficult to improve the communication throughput between the two systems. Accordingly, when specifically the control loop of the control processing system is sped up, a relatively large amount of data needs to be transacted between the intelligent processing system and control processing system, but this is impeded by the hardware restriction due to the above-mentioned problems, resulting in a significant deterioration in the cost-effectiveness evaluation.

SUMMARY OF THE INVENTION

An object of this invention is to provide a processor capable of improving, at well balance and efficient qualities, the processing ability of a multiprocessor or single processor system which is suitable for versatile processing.

The above object is achieved by a processor constituted by a base processor element which includes two CPUs with individual associated local memories, a dual-port RAM (DPR) accessible from the CPUs, and a common bus switch circuit which connects one of the CPUs to a common bus shared by the CPUs.

The inventive processor provides a hardware architecture which allows the two CPUs in the base processor element to operate as if they are a single processor. With attention being paid to the high independency of the control processing system and intelligent processing system based on the data base and sensor information, the control processing system is alloted to the main processing system with the main CPU so that control arithmetics are implemented by the tight-linked parallel processing in unison with other base processor elements, while the intelligent processing system for dealing with overhead processings including the interrupt process, system management and intelligent process is alloted to the background processing system of the main CPU and to the background CPU so as to backup the control processing system of the main CPU. In consequence, the task switching overhead and interrupt events which disturb the parallel processing for control processing system are eliminated as much as possible and the highly independent two processing systems are operated in parallel efficiently, whereby the processing ability of the base processor element is virtually doubled by addition of the processing ability of two CPUs, and also in a multiprocessor system where several base processor elements are linked the total processing ability is doubled as compared with the conventional system and the processing ability can be expanded, at

well balanced grades between the control processing system and intelligent processing system, as the number of base processor elements is increased.

Other objects and features of this invention will be more apparent from the following description of certain preferred embodiments of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram showing the internal arrangement of the base processor element according to this invention and part of the multiprocessor system configured by the base processor element;

Fig. 2 is a diagram showing the switching sequence for the common bus (BPE local bus) shared by the two CPUs in the base processor element according to this invention;

Fig. 3 is a diagram showing the sequence of processes carried out interactively between the two CPUs in the base processor element;

Fig. 4 is a basic logic diagram of the common bus switch according to this invention;

Fig. 5 is a block diagram of the dual-port RAM according to this invention;

Fig. 6 is a block diagram showing another embodiment of this invention applied to an intelligent automatic machine; and

Fig. 7 is a block diagram showing the processor in the system of Fig. 6.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The invention will now be described in detail with reference to the drawings.

In Fig. 1 showing the arrangement of the inventive processor, a base processor element (BPE) 1 for organizing a multiprocessor system includes two CPUs 15 and 16 (CPU0 and CPU1), a dual-port RAM (DPR) 17 as a communication facility between the CPUs 15 and 16, and a common bus switch 24 including a multiplex bus buffer 23 which is controlled consistently by a common bus switch control circuit 22 for connecting any of the CPUs to a BPE local bus 12 shared by the CPUs for establishing communication with other base processor elements (BPEs). The CPUs 15 and 16 have associated with local memories 18 and 20 and local input/output (I/O) units 19 and 21, respectively, so that they can usually operate independently. The dual-port RAM (DPR) 17 which supports the communication between the CPUs features to have communication interrupt lines 32 and 33 to the CPUs 15 and 16, allowing inter-CPU communication with little overhead. The local bus 12 of the base processor element 1 is in connection with a local memory 6 and local I/O unit 7 associated with the base processor element, and at the same time the bus 12 constitutes a common bus line for connection with other base processor elements. Further provided on the local bus 12 is a system bus switch 8 which connects the base processor element to a system bus 14 on which a common system memory 8 and common system I/O unit 10 are connected. The system bus switch 8 implements the bus arbitration process for system bus requests by using an arbitration line 13 so that the common resources on the system bus 14 are used properly and parallel processings are carried out among the base processor elements through the communication process with other base processor elements.

Fig. 5 shows in block diagram the dual-port RAM (DPR) 17, which can be regarded as a memory shared by the CPUs 15 and 16. The DPR 17 includes an arbiter 60 which implements arbitration for the CPUs 15 and 16 in making access to the dual-port RAM (DPR) by using access request signals and access grant signals symbolized as 77 through 80 for each processor, bus switches 61 and 62 which selectively connect the CPU bus 64 or 65 to the internal bus 66 in accordance with the enable signal 75 or 76 issued by the arbiter 60, a decoder 67 which decodes the address and control signals on the internal bus 66 to produce the memory enable signal 81 and interrupt control signals 73 and 74, and flip-flops 68 and 69 which set or reset the interrupt signals 32 and 33 in response to the interrupt control signals 73 and 74 produced by the decoder 67. The dual-port RAM (DPR) has a unique interrupt function for inter-CPU communication, which establishes a register for generating an interrupt to CPU0 and a register for generating an interrupt to CPU1 at certain addresses of the DPR, and at the same time defines these registers to be instruction registers directed to the partner CPUs, thereby implementing instruction reception and interrupt generation at the same time. In an example of instruction issuance by CPU1 to CPU0, the CPU1 initially sets the attribute of the instruction to be executed by CPU0 in its own register and then stores it in the instruction register (interrupt generation register) directed to CPU0 in the DPR, then the decoder 67 knows, by monitoring and decoding the internal bus 66 of the DPR, that the instruction register directed to CPU0 is accessed, and sends the instruction register access signal to CPU0 by using the access signal 73 so that the value of the $\overline{Q}$ signal 70 is latched in the flip-flop 68. Since the flip-flop 68 has been initialized to Q=high and

$\overline{Q}$ =low by the reset signal 72, the above operation causes the flip-flop to produce Q=low and $\overline{Q}$ =high, and the low-active interrupt signal 30 to CPU0 is activated.

Accepting the interrupt, CPU0 again makes reference to the instruction register directed to it in order to fetch an instruction to be executed in its own interrupt service routine, and when it reads out an indicated instruction, the decoder 67 which constantly monitors the access situation provides an instruction register access signal for CPU0 to the flip-flop 68 by using the access signal 73 so as to latch a high $\overline{Q}$ output 70 and output a high to Q. Namely, the interrupt generation line 32 to CPU0 is deactivated. By the foregoing sequence, a series of operations from interrupt generation to interrupt acceptance and operations related to instruction reception on a software basis are carried out concurrently and at a minimal overhead.

Returning to Fig. 1, the common bus switch 24, which selects one of the buses 28 and 29 of CPU0 and CPU1 in the base processor element (BPE) 1 and implements bus switching control for the output to the BPE local bus 12 regarded as a shared bus for CPU0 and CPU1, consists of the aforementioned common bus switching control circuit 22 and the multiplex bus buffer 23 controlled by the circuit 22. The bus switching control, for the case of a master CPU0 and slave CPU1, is implemented by a common bus switching logic made up of a NOR gate 83 and a NAND gate 84 as shown in Fig. 4.

The following describes the unique bus switching control sequence in connection with the timing chart of Fig. 2. Initially, the access right for the two CPU local buses 28 and 29 always resides on the CPU side and it is never infringed by other devices on the bus Ⓐ , Ⓗ . The common bus (BPE local bus 12) access request by CPU0 is always active as shown by Ⓑ , while that of CPU1 is made active when necessary as shown by Ⓘ . Namely, CPU0 normally takes the bus, except only when CPU1 is using it. In the example shown in Fig. 2, the CPU1 raises a common bus access request 87 at Ⓘ a, and CPU0 responds to the request to terminate the instruction process at that time point and output a halt acknowledge 82 immediately when it becomes possible to release the bus right at Ⓓ a and then deactivates the CPU0's common bus access grant signal 85 (driven by the gate 83) at Ⓔ a and releases the bus as shown by Ⓕ a. At Ⓒ a, CPU0 enters the halt state and at the same time at Ⓙ a the CPU1's common bus access grant signal 86 (driven by the gate 84) is activated, causing the bus switch buffer 23 to be selected on the CPU1 side as shown by Ⓚ a, and the bus right is passed to CPU1. When the time comes when CPU1 can release the common bus following the use, it deactivates the CPU1's common bus access request 87 as shown by Ⓘ b . This is immediately followed by the active transition of the CPU0's common bus access grant signal 85 at Ⓔ b, causing the bus switch buffer 23 to be selected on the CPU0 side, and after the bus right is passed to CPU0 its halt acknowledge is reset at Ⓓ b and CPU0 makes a transition from the halt state to the active state at Ⓒ b. In the timing chart, symbols Ⓖ and Ⓛ indicate the active states of CPU0 and CPU1, respectively. In the foregoing master (CPU0) and slave (CPU1) operation, CPU0 is in the halt state for a duration between Ⓕ a and Ⓕ b when the bus right is held by CPU1 and a little while between Ⓕ b and Ⓑ b for the bus switch timing adjustment. Accordingly, CPU1 rather than CPU0 behaves a master's operation from the viewpoint of activity right. A mode is provided to pass the bus right to CPU0 at each data transfer so that CPU0 is not precluded from operating due to a too long halt time. However, as will be described later, for a multiprocessor system using CPU1 as a background CPU for implementing intellignet processings for supporting the main CPU0, adoption of a distributed intelligent base configuration of the functional structure in units of base processor element (BPE) allows each processor to receive most of necessary data in its neighborhood and to have most of data communication through the dual-port RAM (DPR). On this account, transaction of intelligent information among base processor elements (BPEs) is less frequent than information transaction by CPU0 with other base processor element (BPE) for the purpose of tight-linked parallel processing, and therefore loss of processing ability of CPU0 according to this invention is conceivably small. In another case where CPU1 has a fixed role to back up CPU0 and performs system management, it is more efficient for the system to have the CPU0's operation control right committed to CPU1, and the inventive common bus control is said to be suited for the local distributed processing as described above.

The general operation of the foregoing inventive processor will be described in detail with reference to Fig. 3.

Fig. 3 is based on the assumption that CPU0 performs primary control arithmetics, while CPU1 backs up the CPU0 by implementing intelligent processings based on the intelligent base (distributed) and sensor information and also implementing the system management, so that the system operates for local distributed processings. In the case of a multiprocessor configuration, each base processor element (BPE) is assumed to perform a tight-linked parallel processing primarily and

a rough-linked parallel processing on a background basis. Reference number 35 shows the sequence of process by CPUI, and 36, 37 and 38 show the sequence of processes by CPU0. Common resources include a dual-port RAM (DPR) which is a local memory shared by CPU0 and CPUI in the base processor element (BPE), and system common resources on the system bus I4 accessible from all base processor elements (BPEs) in the case of a multiprocessor configuration. Symbols 47, 48, 54 and 59 indicate communications between CPU0 and DPR, while symbols 46, 53, 56 and 58 indicate communications between CPUI and DPR. Similarly, symbols indicate communications between CPU0 and system common resources and between CPUI and system common resources, respectively, and these are regarded as accesses from base processor elements (BPEs) when observed from the system common resources. Symbol 50 indicates an interrupt to CPU0 by utilization of the interrupt function of the dual-port RAM (DPR), and symbol 55 indicates a similar interrupt to CPUI. Symbol 49 indicates a common bus access request signal issued by CPUI to CPU0 and a corresponding bus access grant signal issued by CPU0 in the handshaking operation, and symbol 52 shows the transition of bus right which has once been acquired by CPUI and is now released and returned to CPU0. Symbols 88 and 89 indicate accesses from other BPEs to the system common resources. Symbols 90 and 9I show receptions of external sensor information as part of intelligent of information during the process of CPUI, and similarly symbols 92 and 93 show reception by CPU0 and CPUI of sensor information shared with other BPEs. Concerning the processings performed by CPU0 and CPUI, the CPU0 operates as a main processing system to control, in unison with CPU0s of other base processor elements (BPEs), part of an intelligent machine system, e.g., the arm of a human-type intelligent robot, by executing tight-linked parallel processes 36b and 38b among the CPU0s so that numerous control arithmetic tasks are treated at as high parallelism as possible, and it implements as a background intelligent processing system in unison with CPUI having the process 35 the intelligent processing and system management shown by 36a and 38a in the idle time after a process has been passed to an auxiliary processor, e.g., arithmetic processor, the idle time which occurs during the synchronous process with other base processor element (BPE), and at a process request by interrupt from other base processor element (BPE), CPUI, or system common resources. The intelligent processing system carried out by the base process element (BPE) possesses as a data base information for a further smaller part of the robot arm portion, e.g., information related to muscles, receives related local sensor information as sense information, and executes the intelligent processing related to the muscular portion on the basis of the local function-distributed data base organized as above, thereby backing up the whole control arithmetics carried out by the main processing system.

For the foregoing assumed system, the sequence of processes by CPU0 and CPUI shown in Fig. 3 will be traced briefly. Initially, CPU0 and CPUI are on the way of their processes 36 and 35 shown in Fig. 3. CPUI abruptly develops the need of communication with CPU0, and carries out the operation 46 at time point 39 for writing a communication message in the dual-port RAM (DPR) and writing a command for the message in the instruction register directed to CPU0. This operation generates an interrupt 50 to CPU0, causing the background processing system of CPU0 to access the dual-port RAM (DPR), and communication 47 for necessary information takes place. At time point 40, CPU0 meaninglessly writes and reads data shared with the partner CPU without handshaking to and from the dual-port RAM (DPR). Various sensor information is also processed sequentially or referenced by programs by being entered as interrupts raised by the sensors. Next, CPUI develops the need of communication with a system common resource for the purpose of communication with other BPE, acquires the bus right for the common bus (BPE local bus) I2 at time point 49, has a communication 5I with the system common resource at time point 4I, and after communication returns the bus right back to CPU0 at 52. During the above communication, CPU0 is held in a halt state 37 and, after the halt state has been lifted at 52, it proceeds to a process 38 continuative from the process 36. At the following time point 42, CPUI has a meaningless communication of data shared among the CPUs with the dual-port RAM (DPR), and at 43 CPU0 has an instruction-attached handshake data communication with CPUI in the same way as 39. At time point 44, CPU0 has a communication 57 with a system common resource, and in this case the communication includes one related to intelligent processing during the background process 38a and one for tight-linked parallel processing data related to control arithmetics during the main processing 38b, and these communications do not affect at all the process and operation of CPUI. Although symbol 45 indicates concurrent meaningless communications of CPU0 and CPUI with the dual port RAM (DPR), the arbiter 60 performs proper arbitration control so that the communications take place without any inconvenience to their processes and operations.

When the foregoing intelligent processing system formed of local distributed data bases and control processing system backed up by it is realized by application of the inventive processor, most of the intelligent processes can be executed among the CPUs within the base processor element (BPE) through the dual-port RAM (DPR), with occasional accesses being made to the system common resources for transacting their process results and the intelligent process results provided by other base processor elements (BPEs), whereby the best communication throughput can be accomplished in a natural manner among the communication nodes within the system, and the consequent parallel operations in virtually complete independence between the control processing system and intelligent processing system can surely double the processing ability. Furthermore, by the additional installation of BPEs, the processing ability of the intelligent processing system and control processing system is enhanced proportionally at always well balanced grades between the systems.

According to this embodiment of the present invention, a processor element (base processor element: BPE) which is the fundamental component in constructing a multiprocessor system or single processor system is formed using two CPUs, which seem a single CPU when observed from outside, linked through a dual-port RAM (DPR) with the interrupt function and a common bus shared by the CPUs. The two CPUs are alloted to a high-independence main processing system and background processing system separately, with local information exchange between the CPUs being made through the dual-port RAM (DPR) and communication with other base processor elements (BPEs), in the case of a multiprocessor configuration, being made by way of system common resources on the system bus through a common bus (BPE local bus), whereby the BPE ability is virtually doubled.

When a multiprocessor system is constructed using the inventive processor, where the data base for the background processing system is distributed to each BPE, the background processing system has most of communication within each processor and does not necessitate frequent communication with other processors, and the consequent optimization of communication throughput among the communication nodes enables natural and high-efficiency parallel processings for both of the main processing system and background processing system wihtout a significant influence of local communications on the tight-linked parallel process carried out by the main system. By additional installation of the inventive processor, the processing ability can be enhanced at always well balanced grades between the main and background processing systems.

Next, the expansion of the inventive base processor element (BPE) to a multiprocessor system and an example of the intelligent robot system arranged by use of the multiprocessor system will be described with reference to Fig. 6.

In Fig. 6, reference number 1 denotes base processor elements (BPEs) of this invention. Each base processor element 1 is connected to common buses 14a-14d by a BPE local bus 12. Symbols 99a-99d denote bus arbiters for the respective common buses 14a-14d, and symbols 13a-13d are associated arbitration lines. The common buses 14a-14c are private buses for shared memory systems 9a-9c, respectively, while the bus 14d is a bus for a common I/O unit. The four common buses have individual access rights normally taken by CPU0 (shown by 15 in Fig. 1) which is alloted to the control processing system in the BPE 1, and they are also used by CPU1 (shown by 16 in Fig. 1) which is alloted to the intelligent processing system when necessary. Accordingly, the common buses 14a-14d are roughly considered to be private buses for the control processing system.

CPU0 in each base processor element uses the common buses for communication and carries out necessary control arithmetics as tight-linked parallel processes.

Reference number 101 denotes a hardware unit (communication controller) for controlling parallel processings, and 103 denotes a hardware unit (communication interrupt controller) for controlling interrupt-based communications among base processor elements. The base processor element is provided with a bus switch circuit 102 which supports the system bus 100 as a common bus dedicated to the intelligent processing system and a local bus for CPU1 which is alloted to the intelligent processing system. Placed on the system bus 100 are a common data base 107 and various shared sensors and interfaces. The local bus for CPU1 which is alloted to the control processing system is provided with a floating point arithmetic unit (FPU) 104 used for high-speed arithmetic operations.

The multiprocessor system of this embodiment comprises a total of 14 base processor elements, and the system in modules is provided with an external expansion channel 105 of the type of direct hardware communication unit linkage and an external expansion channel 106 of a software intervention type so that many modules can be linked together. The use of the external expansion channel 105 (hardware direct linkage type) allows high-speed reference to the internal common memories 9a-9c directly from an external module, while the use of

the external expansion channel 106 (software intervention type) allows high-grade, flexible communications based on a certain protocol with other modules.

Using the base processor element arranged as described above, a multiprocessor system which effectively supports the control processing system and intelligent processing system, as described above, can be organized.

The multiprocessor system shown in Fig. 6 is linked with a robot system, as a control object, and it will be described in the following.

The robot system includes robot mechanisms 114a and 114b, a servo unit (including power amplifiers) 111, sensors 113a and 113b equipped on the robot mechanisms 114a and 114b, and another sensor, e.g., television camera, 112. The multiprocessor system of this embodiment performs high-speed coordinate transforming process and dynamical arithmetic process on the part of the control processing system, and delivers the resulting position, speed and acceleration commands to the servo unit 111 by way of a servo interface 110 on the common I/O bus. Conversely, the current operational state of the robot is detected using sensors 5 including encoders, tachogenerators, acceleration sensors and motor current sensors, and the servo unit 111 sends the results of detection as feedback data to the multiprocessor system by way of the servo interface 110. The control processing system in the multiprocessor system uses the feedback data and commands and data provided by the intellignet processing system to calculate subsequent command values to be fed to the servo unit 111. The sensor 112 shared by the two robot mechanisms 114a and 114b sends data to the system bus 100 through a common sensor interface 108, while the sensors 113a and 113b provided individually on the robot mechanisms 114a and 114b send data directly to the intelligent processing systems of individually functioning base processor elements by way of local sensor interfaces 109a and 109b. Based on the information from the sensors 112, 113a and 113b and a common data base 107, the intelligent processing system performs intelligent processings including learning and inferring and provides the results to the control processing system.

According to the above embodiment of this invention, the control processing system performs high-grade dynamical control arithmetics under the support of the high-grade intelligent process, and consequently an intelligent automatic machine control system, such as an intelligent robot, with sensing, inferring and learning functions and abilities of high-speed and high-accuracy motions can be constructed.

According to the above embodiment of this invention, a base processor element, which constitutes the controller for an intelligent automatic machine control system such as an intelligent robot, can implement a control processing system and an intelligent processing system separately, whereby the processing abilities and functions of both systems can be expanded at well balanced grades in the single processor environment or multiprocessor environment. By providing in the base processor element a dual-port RAM dedicated to the local tight-linked communication between the intelligent processing system and control processing system and connecting the two systems directly to the associated CPUs, the possibility of communication disturbance in the control processing system and serious bus contention in the multiprocessor environment can be reduced significantly, whereby communication throughput between the systems and among functions can be made optimal.

By providing a bus switch interface (multiplexer bus buffer and common bus switch control circuit) of a master-slave type for supporting the common bus accessible from the two CPUs in the base processor element so that the base processor in dual-CPU formation is reformed to a single processor, base processors can easily be linked with a plurality of buses, whereby the processing ability and function can be expanded efficiently and uniformly.

According to this invention, as described above, the processing ability of the multiprocessor system or single processor system suitable for general-purpose processings can be enhanced efficiently at well balanced grades.

Moreover, according to this invention, the control processing system, which performs high-grade and high-load dynamical control arithmetic processings under the support of the intelligent processing system based on sensors and intelligent base, has functions of sensing inference and learning, and it accomplishes control for an intelligent automatic machine, such as an intelligent robot, which realizes high-speed and high-accuracy motions.

Claims

I. A processor for constructing a single processor system or multiprocessor system, said processor comprising, within a base processor element (I) constituting said processor, two CPUs (15, 16) with associated local memories (18, 20), a dual-port RAM (DPR) (17) accessible from said CPUs, and a common bus switch circuit (22) for connecting any one of said CPUs to a common bus (12) shared by said CPUs.

2. A processor according to claim I, wherein said dual-port RAM is provided therein with a specific information transmission area, said dual-port RAM having a function of raising a hardware interrupt, in response to an access to said area by any one of said CPUs, to said bus accessing CPU or to another CPU.

3. A processor according to claim I, wherein said common bus switch implements common bus access control in master-slave mode, in which said two CPUs are ranked such that one CPU operates as a main CPU and another CPU operates as a background CPU, said main CPU normally having access right to said common bus, while said background CPU issuing a bus access request signal to said main CPU when need of access to said common bus has arisen, said main CPU, upon recognition of said bus access request signal, entering a halt state at a time point when said background CPU is ready to access to said common bus and, at the same time, issuing an access grant signal to said background CPU to allow said background CPU to make access to said common bus, said halt state of said main CPU being lifted in response to the removal of said bus access request signal upon completion of common bus access by said background CPU, with the common bus access right being returned to said main CPU.

4. A processor according to claim I or 2, wherein said dual-port RAM is provided in a specific information transmission area thereof with interrupt generating registers on itself and interrupt generating flip-flops (68, 69) for generating an interrupt to each of said two CPUs, an instruction, which is written by one CPU into one of said interrupt generating registers corresponding to a CPU appointed to execute an instruction processing, being decoded on a hardware basis, a corresponding one of said interrupt generating flip-flops being set to activate an interrupt line (32, 33) so that an interrupt to said appointed CPU is generated, said CPU upon receiving said interrupt making reference, in an interrupt service routine, to said interrupt generating register directed thereto where said instruction has been written to read out said instruction, which is decoded on a hardware basis, causing said flip-flop to be reset and said interrupt line to said appointed CPU to be deactivated, then reception of interrupt being completed.

5. A processor for constructing a single processor system and controlling a machine (II4), said processor comprising, within a base processor element (I) constituting said processor, two CPUs (I5, I6) with associated local memories (I8, I9) for implementing separately an intelligent processing system and a control processing system of said machine, a dual-port RAM (I7) accessible from said CPUs, and a common bus switch circuit (22) for connecting any one of said CPUs to a common bus (I2) shared by said CPUs.

6. A processor for constructing a multiprocessor system and controlling a machine (II4), said processor comprising, within a base processor element (I) constituting said processor, a first CPU (I6) having a local memory (20) and implementing a process for a control processing system of said machine, a second CPU (I5) having a local memory (I8) and implementing a process for an intelligent processing system of said machine, a dual-port RAM (I7) accessible from said CPUs, a common bus switch circuit (22) for connecting any one of said CPUs to a common bus (I2) shared by said CPUs, and a common bus connected to an intelligent processing controller and a control processing controller (III) of said machine.

7. A processor for constructing a multiprocessor system and controlling a machine (II4), said processor comprising, within a base processor element (I), a first CPU (I6) having a local memory (20) and implementing a process for a control processing system of said machine, a second CPU (I5) having a local memory (I8) and implementing a process for an intelligent processing system of said machine, a dual-port RAM (I7) accessible from said CPUs, and a common bus switch circuit (22) for connecting said CPUs to a common bus, said first CPU being connected to a common bus (I4) of said control processing system, while said second CPU being connected to a common bus (I00) of said intelligent processing system.

FIG. 1

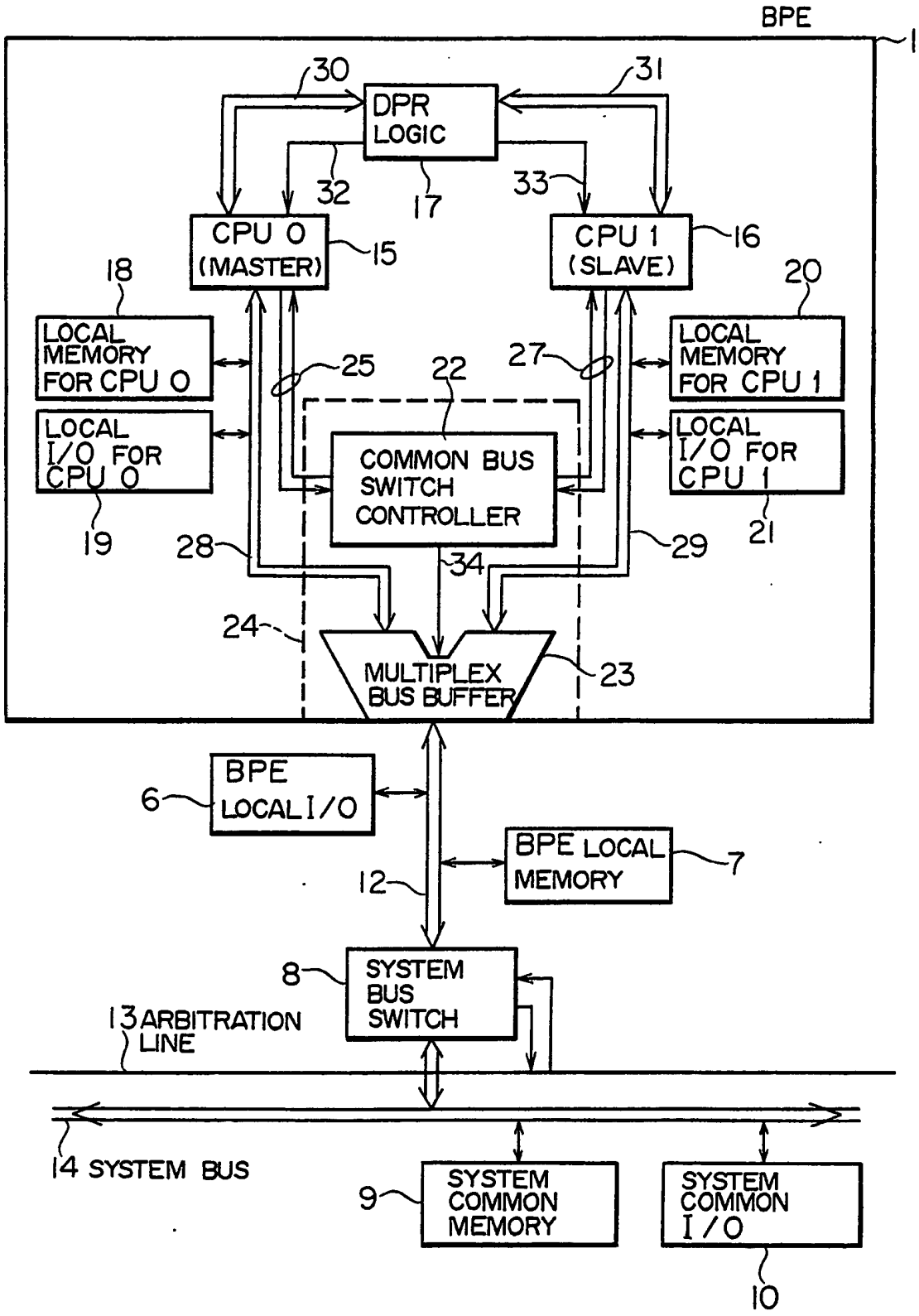

BPE
1
30
31
DPR LOGIC
32
17
33
CPU 0 (MASTER)
15
CPU 1 (SLAVE)
16
18
20
LOCAL MEMORY FOR CPU 0
25
22
27
LOCAL MEMORY FOR CPU 1
LOCAL I/O FOR CPU 0
COMMON BUS SWITCH CONTROLLER
LOCAL I/O FOR CPU 1
19
28
29
21
34
24
MULTIPLEX BUS BUFFER
23
BPE LOCAL I/O
6
BPE LOCAL MEMORY
7
12
8
SYSTEM BUS SWITCH
13 ARBITRATION LINE
14 SYSTEM BUS
9
SYSTEM COMMON MEMORY
SYSTEM COMMON I/O
10

FIG. 2

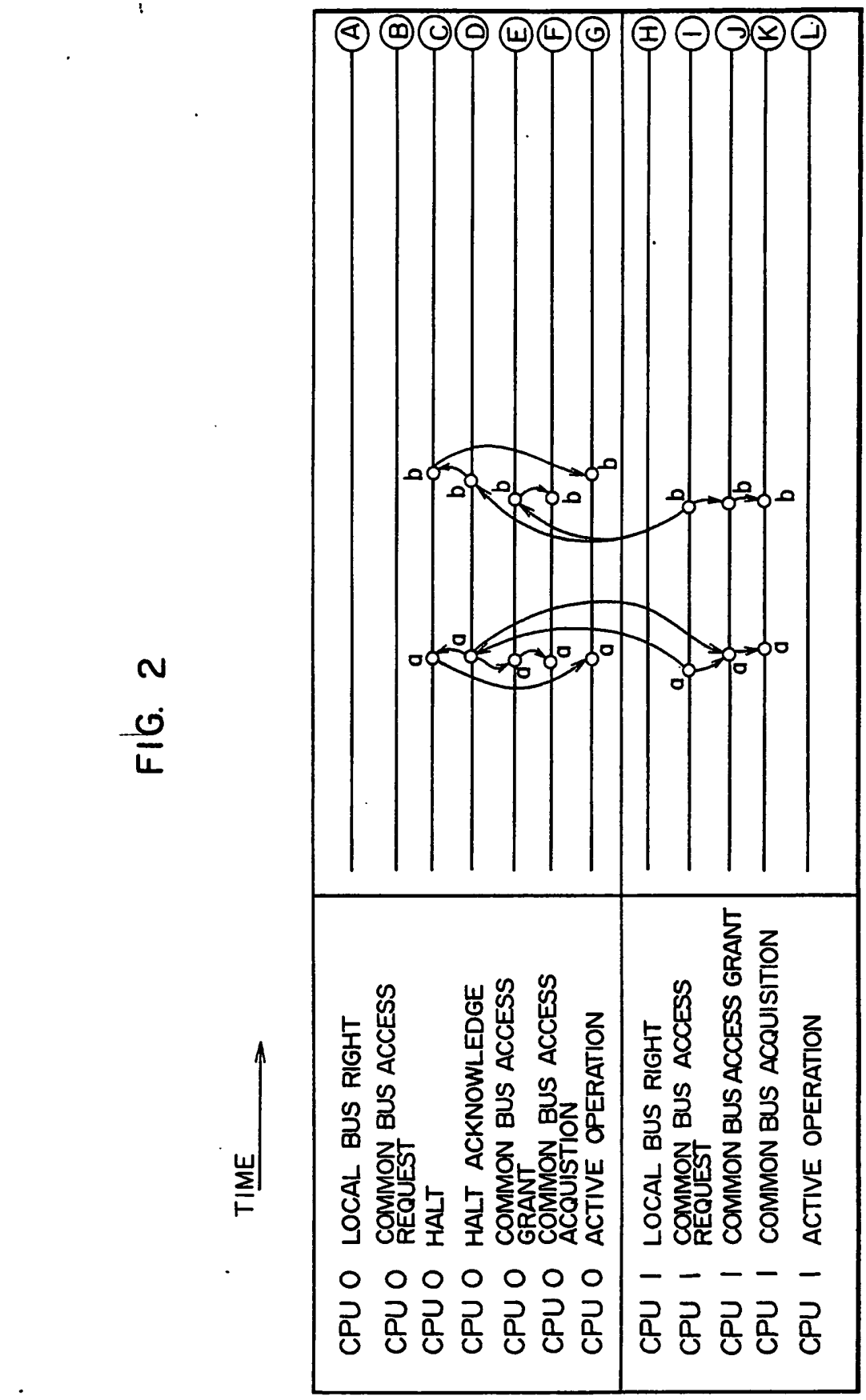

TIME
CPU 0 LOCAL BUS RIGHT
CPU 0 COMMON BUS ACCESS REQUEST
CPU 0 HALT
CPU 0 HALT ACKNOWLEDGE
CPU 0 COMMON BUS ACCESS GRANT
CPU 0 COMMON BUS ACCESS ACQUISITION
CPU 0 ACTIVE OPERATION
CPU 1 LOCAL BUS RIGHT
CPU 1 COMMON BUS ACCESS REQUEST
CPU 1 COMMON BUS ACCESS GRANT
CPU 1 COMMON BUS ACQUISITION
CPU 1 ACTIVE OPERATION
A
B
C
D
E
F
G
H
I
J
K
L
a
b

FIG. 3

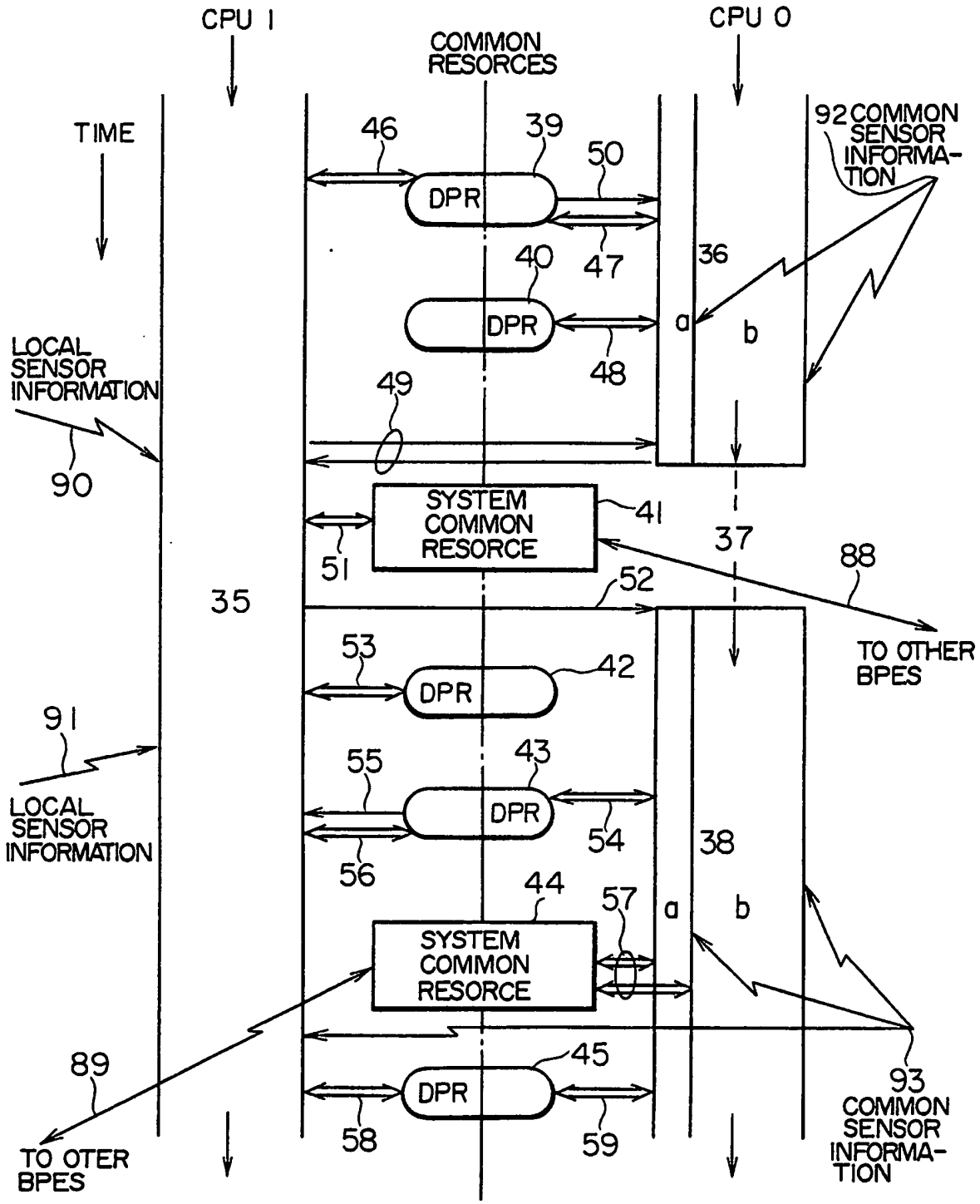

CPU 1
COMMON RESORCES
CPU 0
TIME
46
39
50
DPR
92 COMMON SENSOR INFORMA-TION
40
47
36
DPR
a
b
LOCAL SENSOR INFORMATION
48
49
90
SYSTEM COMMON RESORCE
41
51
37
88
35
52
TO OTHER BPES
53
42
DPR
91
55
43
LOCAL SENSOR INFORMATION
DPR
54
38
56
44
57
a
b
SYSTEM COMMON RESORCE
89
45
93 COMMON SENSOR INFORMA-TION
DPR
58
59
TO OTER BPES

FIG. 4

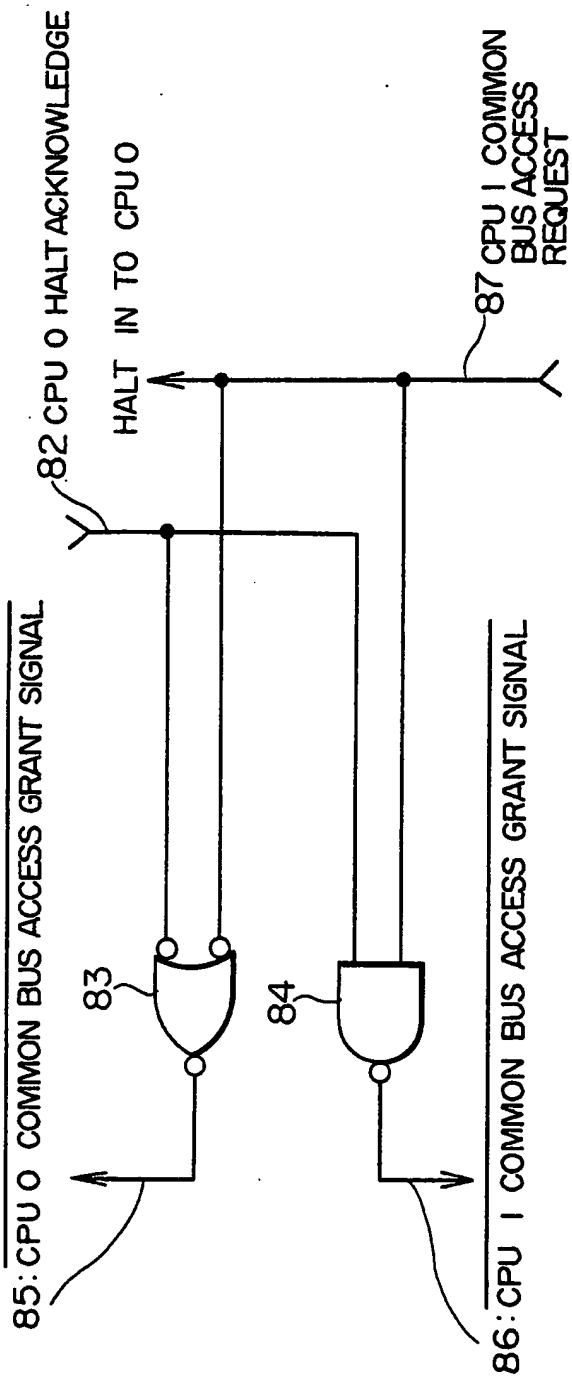
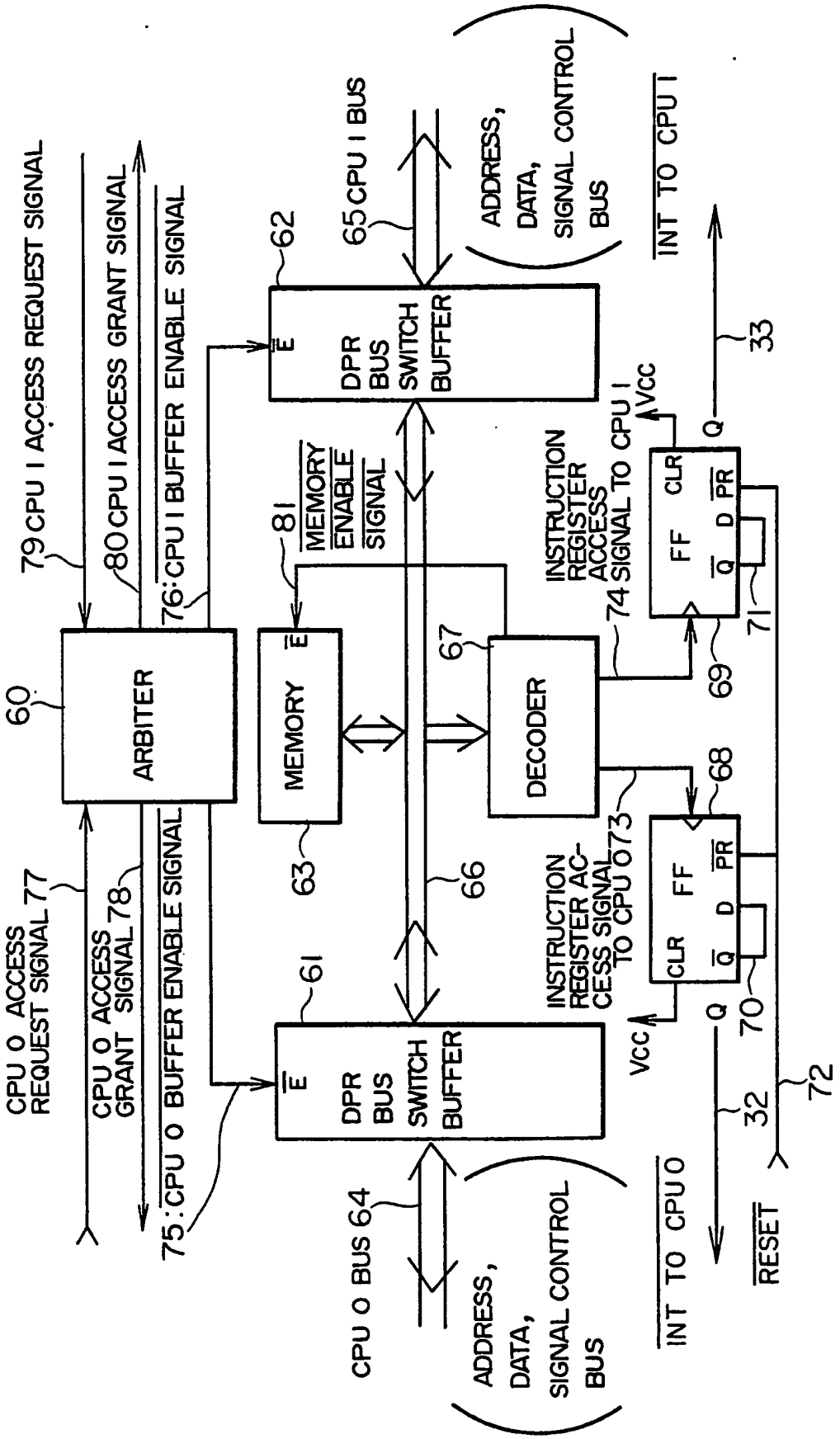

82 CPU 0 HALT ACKNOWLEDGE
HALT IN TO CPU 0
87 CPU 1 COMMON BUS ACCESS REQUEST
83
84
85: CPU 0 COMMON BUS ACCESS GRANT SIGNAL
86: CPU 1 COMMON BUS ACCESS GRANT SIGNAL

FIG. 5


79 CPU I ACCESS REQUEST SIGNAL
80 CPU I ACCESS GRANT SIGNAL
76: CPU I BUFFER ENABLE SIGNAL
CPU O ACCESS REQUEST SIGNAL 77
CPU O ACCESS GRANT SIGNAL 78
75: CPU O BUFFER ENABLE SIGNAL
60
ARBITER
62
65 CPU I BUS
DPR BUS SWITCH BUFFER
ADDRESS, DATA, SIGNAL CONTROL BUS
INT TO CPU I
33
81 MEMORY ENABLE SIGNAL
MEMORY
63
67
DECODER
66
INSTRUCTION REGISTER ACCESS 74 SIGNAL TO CPU I
FF CLR D PR Q
Vcc
69
71
68
INSTRUCTION REGISTER ACCESS SIGNAL TO CPU 073
61
DPR BUS SWITCH BUFFER
CPU O BUS 64
ADDRESS, DATA, SIGNAL CONTROL BUS
Vcc
70
32
72
INT TO CPU O
RESET

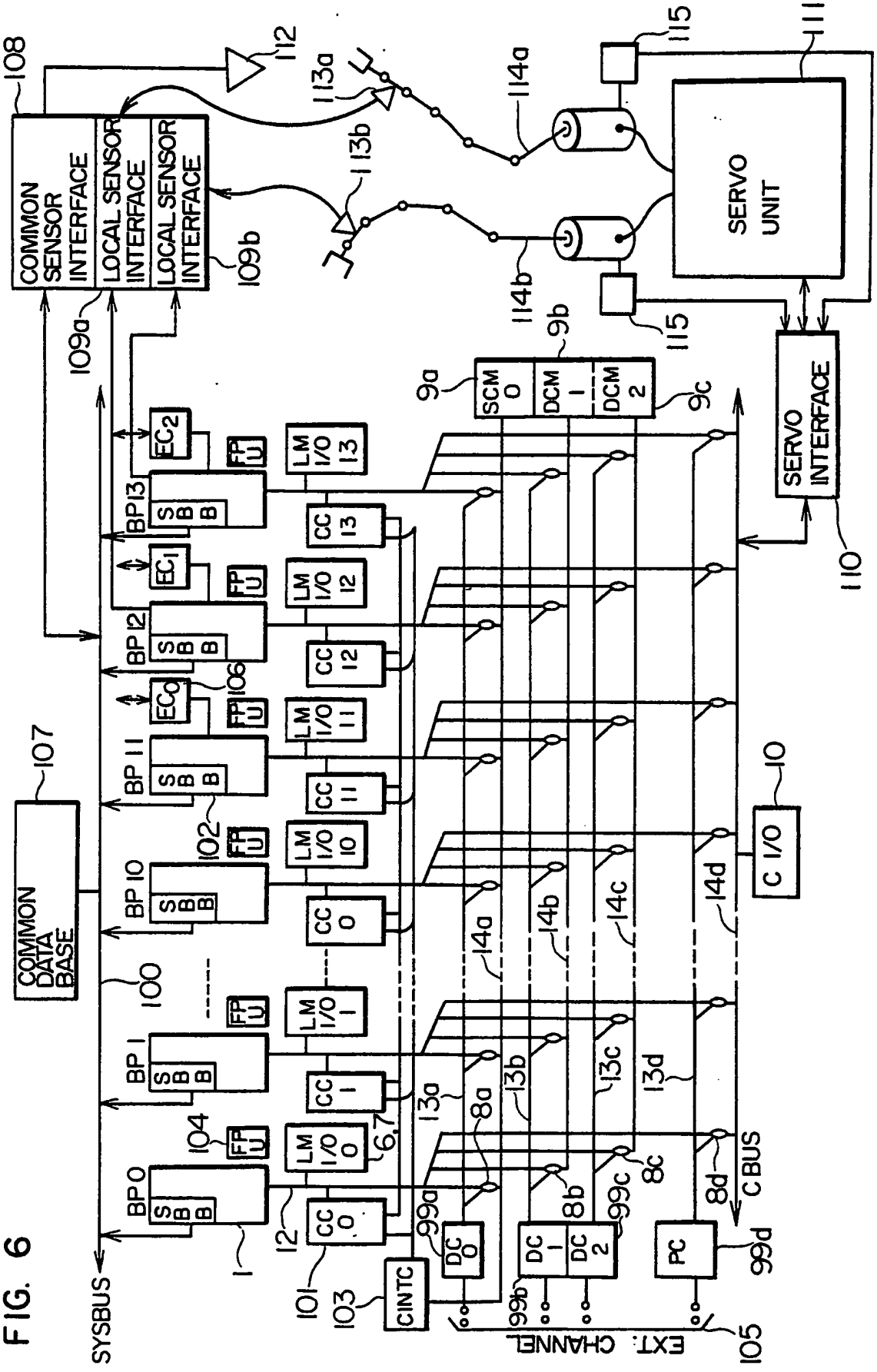
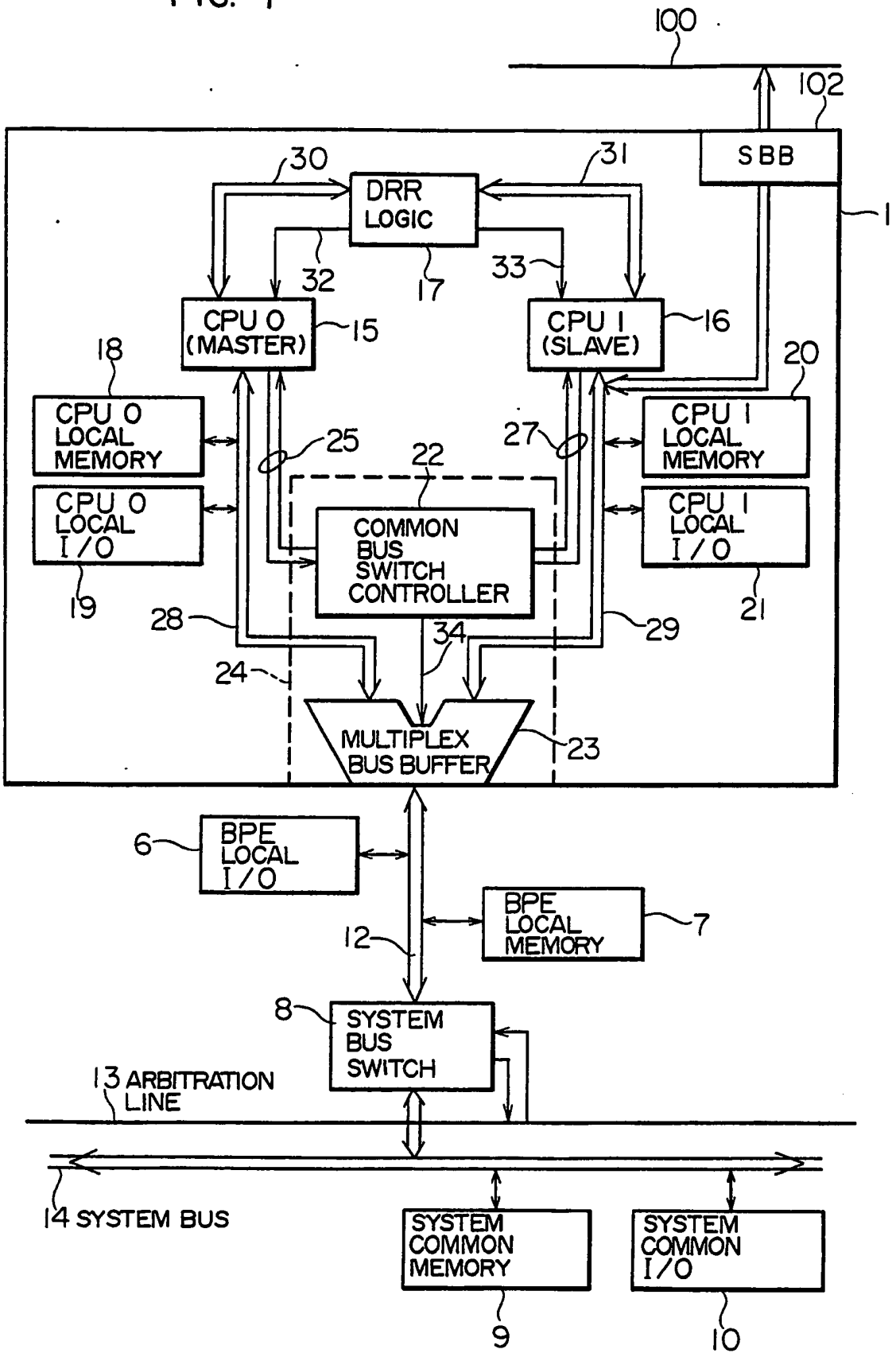

COMMON SENSOR INTERFACE
LOCAL SENSOR INTERFACE
LOCAL SENSOR INTERFACE
SERVO UNIT
SERVO INTERFACE
COMMON DATA BASE
SCM 0
DCM 1
DCM 2
BP 0
BP 1
BP 10
BP 11
BP 12
BP 13
SBB
FP U
LM I/O 0
LM I/O 1
LM I/O 10
LM I/O 11
LM I/O 12
LM I/O 13
CC 0
CC 1
CC 11
CC 12
CC 13
EC0
EC1
EC2
CINTC
DC 0
DC 1
DC 2
PC
C I/O
SYSBUS
C BUS
EXT. CHANNEL
108
109a
109b
112
113a
113b
114a
114b
115
111
110
107
100
102
106
104
101
103
12
1
67
13a
13b
13c
13d
14a
14b
14c
14d
8a
8b
8c
8d
9a
9b
9c
99a
99b
99c
99d
105
10

FIG. 6

FIG. 7


100
102
SBB
1
30
31
DRR LOGIC
32
17
33
CPU 0 (MASTER)
15
CPU 1 (SLAVE)
16
18
20
CPU 0 LOCAL MEMORY
CPU 1 LOCAL MEMORY
25
27
22
CPU 0 LOCAL I/O
COMMON BUS SWITCH CONTROLLER
CPU 1 LOCAL I/O
19
21
28
29
34
24
MULTIPLEX BUS BUFFER
23
6
BPE LOCAL I/O
BPE LOCAL MEMORY
7
12
8
SYSTEM BUS SWITCH
13 ARBITRATION LINE
14 SYSTEM BUS
SYSTEM COMMON MEMORY
SYSTEM COMMON I/O
9
10